

WHAT IS CLAIMED IS:

1. A MAP decoder, comprising:

a backward processor that calculates first resultant values of an L-symbol length sequence and second resultant values of a W-symbol length sequence;

a forward processor that calculates third resultant values;

a memory that stores the second resultant values in a first order and outputs the second resultant values in a second order that is the reverse of the first order; and

an output determination module that determines output values of a received sequence using the third resultant values and the outputted second resultant values, wherein

the L-symbol length sequence and the W-symbol length sequence are portions of the received sequence,

the first, second, and third resultant values are state probability values,

the calculations of the first and third resultant values overlap in time, and

the calculation of the first resultant values is performed after the calculation of the second resultant values is completed.

2. The MAP decoder of claim 1, wherein the memory writes a plurality of groups of the second resultant values by alternately using increasing and decreasing sequential addresses to write subsequent groups of the second resultant values.

3. The MAP decoder of claim 1, wherein the memory repeatedly writes newly calculated second resultant values and reads the stored second resultant values by alternately using: (1) increasing sequential addresses for the write operation and decreasing sequential addresses for the read operation and (2) decreasing sequential addresses for the write operation and increasing sequential addresses for the read operation, as the write and read operations are applied to each of a plurality of groups of the second resultant values.

4. The MAP decoder of claim 1, wherein:

the received sequence has a symbol length of M symbols;

each of the first resultant values corresponds to an input value of the L -symbol length sequence;

each of the first resultant values is calculated in a sequential order that is the reverse of the order of receipt of the corresponding input value; and

a first calculated value in the ordered sequence of the first resultant values corresponds to the i^{th} symbol received in the received sequence, where i is identified by the equation $i = W + L + (M \bmod W)$ and $(M \bmod W)$ is modulo division providing the remainder of the division.

5. The MAP decoder of claim 1, wherein:

the received sequence has a symbol length of M symbols;

each of the second resultant values corresponds to an input value of the W -symbol length sequence;

each of the second resultant values is calculated in a sequential order that is the reverse of the order of receipt of the corresponding input value; and

a first calculated value in the ordered sequence of the second resultant values corresponds to the j^{th} symbol received in the received sequence, where j is identified by the equation $j = (M \bmod W)$ and $(M \bmod W)$ is modulo division providing the remainder of the division.

6. The MAP decoder of claim 1, wherein:

the received sequence has a symbol length of M symbols;

each of the third resultant values corresponds to an input value of the W -symbol length sequence;

each of the third resultant values is calculated in the order of receipt of the corresponding input value; and

a last calculated value in the ordered sequence of the third resultant values corresponds to the k^{th} symbol received in the received sequence, where k is identified by the equation $k = (M \bmod W)$ and $(M \bmod W)$ is modulo division providing the remainder of the division.

7. The MAP decoder of claim 3, wherein:

each of the output values corresponds to an input value of the W -symbol length sequence; and

the output determination module outputs each of the output values in the same order as the order of receipt of the corresponding input value of the W -symbol length sequence.

8. A method of performing a MAP turbo decoding, comprising:

calculating first resultant values of an L-symbol length sequence;

calculating second resultant values of a W-symbol length sequence;

calculating third resultant values of the W-symbol length sequence;

storing the second resultant values in a first order and outputting the stored second resultant values in a second order that is the reverse of the first order; and

outputting decoded values of a received sequence using the third resultant values and the outputted second resultant values, wherein

the L-symbol length sequence and W-symbol length sequence are portions of the received sequence,

the first, second, and third resultant values are state probability values,

the calculations of the first and third resultant values overlap in time, and

the calculation of the first resultant values is performed after the calculation of the second resultant values is completed.

9. The method of claim 8, further comprising writing a plurality of groups of the second resultant values by alternately using increasing and decreasing sequential addresses to write subsequent groups of the second resultant values.

10. The method of claim 8, further comprising:

repeatedly writing newly calculated second resultant values and reading the stored second resultant values by alternately using: (1) increasing sequential addresses for the write operation and decreasing sequential addresses for the read operation and (2) decreasing sequential addresses for the write operation and increasing sequential addresses for the read operation, as the write and read operations are applied to each of a plurality of groups of the second resultant values.

11. The method of claim 8, wherein:

the received sequence has a symbol length of M symbols;

each of the first resultant values corresponds to an input value of the L-symbol length sequence;

each of the first resultant values is calculated in a sequential order that is the reverse of the order of receipt of the corresponding input value; and

a first calculated value in the ordered sequence of the first resultant values corresponds to the i^{th} symbol received in the received sequence, where i is identified by the equation $i = W + L + (M \bmod W)$ and $(M \bmod W)$ is modulo division providing the remainder of the division.

12. The method of claim 8, wherein:

the received sequence has a symbol length of M symbols;

each of the second resultant values corresponds to an input value of the W-symbol length sequence;

each of the second resultant values is calculated in a sequential order that is the reverse of the order of receipt of the corresponding input value; and

a first calculated value in the ordered sequence of the second resultant values corresponds to the j^{th} symbol received in the received sequence, where j is identified by the equation $j = (M \text{ mod } W)$ and $(M \text{ mod } W)$ is modulo division providing the remainder of the division.

13. The method of claim 8, wherein:

the received sequence has a symbol length of M symbols;

the W-symbol length sequence is a portion of the received sequence;

each of the third resultant values corresponds to an input value of the W-symbol length sequence;

each of the third resultant values is calculated in the order of receipt of the corresponding input value; and

a last calculated value in the ordered sequence of the third resultant values corresponds to the k^{th} symbol received in the received sequence, where k is identified by the equation $k = (M \text{ mod } W)$ and $(M \text{ mod } W)$ is modulo division providing the remainder of the division.

14. The method of claim 8, wherein:

each of the decoded values corresponds to an input value of the W-symbol length sequence; and

the decoded values are output in the same order as the order of receipt of the corresponding input value of the W-symbol length sequence.

15. A method of turbo-decoding a received sequence using a Maximum A Posteriori (MAP) algorithm, comprising:

performing a learning by a backward processor for a predetermined length;

calculating and storing first resultant values obtained by the backward processor;

calculating second resultant values by a forward processor that overlaps in time with the learning; and

determining a decoding symbol output using the second resultant values and the stored first resultant values.

16. The turbo-decoding method of claim 15, wherein if a processing length of the backward or forward processor is W, a length of learning is L, a remainder obtained by dividing a length of a received sequence by W is W_0 , and N is an integer not less than 1,

the learning is performed by the backward processor on sequential symbol portions of the received sequence identified by the range $W_0 + NW + L$ to $W_0 + NW$;

the first resultant values calculated by the backward processor on sequential symbol portions identified by the range of the received sequence W_0+NW to $W_0+(N-1)W$ are stored;

the second resultant values calculated by the forward processor on the sequential symbol portions of the received sequence identified by $W_0+(N-1)W$ to W_0+NW ; and

the decoding symbol determination is performed with the second resultant values and the first resultant values based on the sequential symbol portions of the received sequence identified by the range $W_0+(N-1)W$ to W_0+NW .

17. The turbo-decoding method of claim 15, wherein if a processing length of the backward or forward processor is W , a length of learning is L , a remainder obtained by dividing a length of the received sequence by W is W_0 , and N is an integer equal to 0;

the learning is performed by the backward processor on sequential symbol portions of the received sequence identified by the range W_0+NW+L to W_0+NW ;

the first resultant values calculated by the backward processor on sequential symbol portions of the received sequence identified by the range W_0+NW to 0 are stored; and

the second resultant values calculated by the forward processor on the sequential symbol portions of the received sequence identified by the range 0 to W_0+NW are calculated during a period overlapping in time simultaneously the learning performed in a next window.

18. The turbo-decoding method of claim 15, wherein the first resultant values are written through one port of a dual-port RAM (DPRAM) and are read out through another port thereof.

19. The method of claim 15, further comprising writing a plurality of groups of the first resultant values by alternately using increasing and decreasing sequential addresses to store subsequent groups of the first resultant values.

20. The method of claim 15, further comprising:

repeatedly storing newly calculated first resultant values and reading the stored second resultant values by alternately using: (1) increasing sequential addresses for the store operation and decreasing sequential addresses for the read operation and (2) decreasing sequential addresses for the store operation and increasing sequential addresses for the read operation, as the write and read operations are applied to each of a plurality of groups of the first resultant values.

21. The method of claim 15, further comprising:

outputting a plurality of the decoding symbol outputs as decoded values, wherein each of the decoded values corresponds to an input value of a W-symbol length sequence; and

the decoded values are output in the same order as the order of receipt of the corresponding input value of the W-symbol length sequence.

09972953-101501